



UNIVERSIDAD DE LOS ANDES

FACULTAD DE INGENIERÍA

ESCUELA DE INGENIERÍA DE SISTEMAS

Diseñar, desarrollar e implementar un sistema de información
www.bdigital.ula.ve
web para la gestión de datos de ULA *Startups*.

Por

Br. Anny Vanessa Gil Calderon

Tutor: Prof. Gerard Páez Monzón

Febrero 2017

©2017 Universidad de Los Andes Mérida, Venezuela

C.C. Reconocimiento



UNIVERSIDAD DE LOS ANDES

FACULTAD DE INGENIERÍA

ESCUELA DE INGENIERÍA DE SISTEMAS

Trabajo de grado presentado como requisito para

optar por el título de Ingeniero de Sistemas

www.bdigital.ula.ve

Diseñar, desarrollar e implementar un sistema de información

web para la gestión de datos de ULA *Startups*.

Por

Br. Anny Vanessa Gil Calderon

Tutor: Prof. Gerard Páez Monzón

Febrero 2017

©2017 Universidad de Los Andes Mérida, Venezuela

C.C. Reconocimiento

Diseñar, desarrollar e implementar un sistema de información web para la gestión de datos de ULA *Startups*.

Br. Anny Vanessa Gil Calderon

Proyecto de Grado — Sistemas de Computación, 67 páginas

Resumen:

Los sistemas de gestión son importantes para administrar sistemas web de formación debido a que dan soporte a un conjunto de utilidades para el seguimiento de los involucrados. Ofreciendo al usuario una interfaz agradable y de fácil uso, y la funcionalidad necesaria para que funcione de manera eficiente.

En este trabajo se presenta el desarrollo de un sistema de información web que darán soporte a los procesos de una aceleradora de startups, manejo de cursos, permite el seguimientos de los emprendimientos, además facilita la gestión del contenido y programación de cursos y eventos propios del desarrollo de una aceleradora, manejo de perfiles como inversionista, startups y administrador. Facilita el progreso de las startups a través de un mínimo producto viable, haciendo uso de herramientas tecnológicas.

Se desarrolló utilizando UML 2.0 (Unified Modeling Language) para el modelado y como guía en su desarrollo, el método Watch (versión 2004) para el desarrollo de aplicaciones empresariales, realizando 3 iteraciones.

Palabras Clave: Sistema de Información Web, Aceleradora, MVP, Startups, Emprendimiento.

www.bdigital.ula.ve

C.C. Reconocimiento

Dedicatoria

*A mi madre, mi padre y mi familia quienes me han acompañado y apoyado en todo momento,
y son un ejemplo digno de seguir.*

Índice

Resumen	i
Dedicatoria	ii
Índice	iii
Capítulo 1.	8
Introducción.	8
Antecedentes.	11
Planteamiento del problema.	13
Objetivo General.	14
Objetivos específicos.	14
Metodología.	15
Recolección de historias de los usuarios.	15
1.5.2 Pila del producto.	16
1.5.3 Iteraciones.	16
1.5.4 Análisis del producto mínimo viable.	17
1.5.5 Retroalimentación.	18

Planificación.	18
Capítulo 2.	21
2.1 Especificaciones de ULA Startups.	21
2.2 La estrategia de Plataforma web ULA Startups.	22
2.2.1 ULA startups desde el aula de clases.	22
2.2.2 ¿Cómo se implementa ULA startups?	22
2.2 Especificaciones técnicas de la plataforma ULA startups.	23
2.2.1 Plataforma web.	23
2.2.1 Aceleradora de startups.	23
2.2.2 Arquitectura cliente servidor.	24
2.2.4 Modelado de base de datos.	24
Capítulo 3.	26
3.1 Requerimientos del sistema.	26
3.1.1 Requerimientos funcionales.	26
3.1.2 Requerimientos no funcionales.	26
3.1.3 Requerimientos de calidad.	27
3.1.4 Requerimientos de seguridad.	27
3.2 Modelado del sistema.	27

3.2.1 Actores.	28
3.3 Casos de uso.	33
3.3.1 Actor Usuario.	33
3.3.1.1 Registrar.	34
3.3.1.2 Ver pagina principal.	35
3.3.2 Actor Administrador.	36
3.3.2.1 Cambiar perfil.	36
3.3.2.2 Gestionar startups.	37
3.3.2.3 Gestionar Inversionistas.	39
3.3.2.4 Gestionar Ponencias.	40
3.3.2.5 Gestionar Eventos.	41
3.3.2.6 Gestionar Pagina Principal.	43
3.3.2.7 Admitir Startup.	44
3.3.2.8 Admitir Inversionista.	45
3.3.2.9 Ver y responder mensajes.	46
Capítulo 4.	47
4.1 Justificación y diseño.	47
4.2 Diagrama de despliegue.	49

4.3 Arquitectura de software.	49
4.4 Modelo relacional de la base de datos.	51
4.5 Diagrama de entidad relación extendido de la base de datos.	53
4.6 Diseño de la interfaz de usuarios.	54
Capítulo 5.	55
5.1 Desarrollo de ULA Startups.	55
5.2 Pruebas del sistema.	57
Capítulo 6.	59
6.1 Conclusiones.	59
6.2 Recomendaciones.	60
Capítulo 7.	62
Capítulo 8.	64
8.1 Definiciones técnicas.	64
8.1.1 HTML	64
8.1.2 Navegador Web.	65
8.1.3 Javascript.	65
8.1.4 PHP.	65

8.1.5 MVC.	65
8.1.6 Framework.	66
8.1.7 UML.	66
8.1.8 SQL.	67
8.1.9 MySQL.	67

www.bdigital.ula.ve

Capítulo 1

1.1 Introducción

La universidad venezolana se abre a tiempos de cambio. Actualmente, uno de los aspectos más relevantes en el campo educativo es la creatividad en períodos cruciales para consolidar proyectos que tengan un gran significado para todos. Por ello, la Universidad de Los Andes consagrada en cumplir su papel histórico de promover la transformación en la gestión del conocimiento y del pensamiento, cada día ha estado más comprometida en su rol para lograr la verdadera dimensión y pertinencia social. Los estudiantes, profesores, empleados y autoridades son todos protagonistas del aprendizaje en una organización más humanizada y avanzada, representando una reserva de talento, capacidad pedagógica, moral y valores éticos. De esta manera, se aspira a la nueva universidad por vía de ideas bien concebidas y administradas con inteligencia dentro de un marco de pluralidad y diversidad; generando conocimientos e

incorporando y creando nuevas tecnologías destinadas a una sociedad más globalizada y competitiva.

A día de hoy y conscientes de las necesidades de la sociedad actual y como una forma de dar soluciones efectivas, se impone un proceso de transformación de la universidad, en una época donde la innovación y el emprendimiento se unen para dar respuesta a los problemas. En esta nueva perspectiva, se gestionan programas y proyectos que permiten revitalizarla como organización; asimismo, los estudiantes son capaces de desarrollar tecnologías mientras aplican los conocimientos obtenidos en su carrera universitaria, construyendo así un producto escalable y de interés común para muchos. En este proceso de cambio, surge la necesidad de orientación y asesoría en todos los aspectos que conllevan a la ejecución de una idea. Otro aspecto clave está representado por todas las herramientas que permitan organizar, visualizar cada etapa y acelerar el desarrollo del proyecto. Lo planteado tiene implicaciones importantes, con especial atención en la construcción de una plataforma tecnológica para el emprendimiento universitario con software especializado en la asesoría legal, valoración de métricas, modelo de negocios, conexión con otros inversionistas o innovadores y gestión de tareas.

Esta combinación de herramientas proporciona apoyo a la creación de un ambiente de innovación y desarrollo, dando soporte al programa ULA *startups*; este último constituyéndose como el pilar fundamental de la gestión y aceleración de las ideas cuyo génesis sea desde la

universidad. A continuación se muestra los proyectos de grados que conforman la plataforma en general, han sido integrados como parte de una sola estructura:

1. Diseñar, Desarrollar e Implementar un Sistema de Información Web para la Gestión de Datos de ULA Startups (Gil, A; 2017).
2. Aplicación en Javascript para el Desarrollo Startups (Cuevas, R; 2017).
3. Aplicación Web para la Gestión de Documentos Legales de las Startups (Espinoza, L; 2017).
4. GeMVP. Diseño y Construcción de un Gestor de Producto Mínimo Viable para una Startup (Briceño, G; 2017).
5. Desarrollo e Implementación de un Algoritmo para Valorar y Jerarquizar un Conjunto de Emprendimientos Tecnológicos de Arranques (Ramírez, F; 2017).

ULA *startups* podría ser el resultado de la ruptura de los paradigmas tradicionales de la universidad, así como la integración de todos los sectores de la comunidad en los procesos de cambio y concepción de la nueva universidad.

1.2 Antecedentes

En el 2007 una actividad en la ULA, específicamente con el propósito de mejorar el aprendizaje de las asignaturas, surge y se agrupa bajo el nombre de radixPuntoEDU creando una estrategia educacional bautizada R.A.I.S que consistía en Reproducir el Ambiente de Innovación en el Salón de clase (Entrevista personal con los profesores Beatriz Sandia y Gerard Páez).

www.bdigital.ula.ve

La estrategia RAIS nace para transformar las asignaturas, su objetivo es simular un ambiente de innovación que promueve una cultura de emprendimiento en los estudiantes, para adaptarlos más a la realidad mundial de la disponibilidad de la información y herramientas. El salón de clase se convierte en un espacio donde un conjunto de compañías se conforman por un grupo de estudiantes, dirigidas y evaluadas por el profesor, donde los conocimientos adquiridos de la asignatura son puestos en práctica en el desarrollo del producto. (Beatriz Sandia y Gerard Páez).

Hoy en día en la Facultad de Ingeniería de la Universidad de Los Andes un buen número de profesores implementan la estrategia RAIS en el salón de clase y se generan productos al que posteriormente son expuestos en expoRAIS una vez al año. Este crecimiento tanto formativo

C.C. Reconocimiento

como creativo ha generado en el grupo Radix Punto EDU la necesidad de desarrollar un nuevo programa de formación, con el nombre de *ULA Startups*, que consiste en llevar estos productos RAIS al mercado. *ULA Startups* es una aceleradora de ideas y talentos orientados a la cultura *Startup*. *ULA Startups* ofrece un ambiente de emprendimiento universitario para impulsar los productos que nacen en los salones de clases RAIS con el fin de encontrar inversionistas que permitan su realización.

Existen otros sistemas web de gestión de aceleradoras conocidas entre la que destaca Telanto(2012) [5], por la similitud que tiene con *ULA Startups*, en cuanto a la incorporación de las universidades para el nacimiento de nuevas ideas a emprender, bajo un cultura de emprendimiento. Esta plataforma facilita la gestión de la aceleradora ya que permite la interacción de las *startups* con los expertos, seguimiento de los emprendimientos y notificación de eventos, entre otras funcionalidades.

Si bien esta plataforma facilita la gestión y funcionamiento tecnológico de una aceleradora, está enfocada en mejorar la experiencia de los emprendedores e inversores y la incorporación de las universidades en el proceso de desarrollo de ideas, no cuenta con los fundamentos de *ULA Startups*, en lo que además de lo ya señalado, requiere de un mecanismo adaptado a profesores RAIS y productos RAIS, que vienen con una base en la cultura del emprendimiento, desde los salones de clase, a concretar estos productos, contando con un adiestramiento personalizado y

herramientas que facilitaran su objetivo de encontrar inversionistas, y así ofrecer la experiencia necesaria para futuros profesionales y emprendedores.

1.3 Planteamiento del problema

ULA *Startups* ofrece un ambiente de desarrollo de emprendimiento que plantea una formación intensiva y continua, orientada hacia la ejecución de productos y finalmente obtener el financiamiento por parte de inversores. Esta formación se basa en cursos, ponencias y eventos donde los estudiantes deben evidenciar su nivel de preparación al ejecutar el producto o idea y obtener un producto mínimo viable para el mercado. Además, hay un seguimiento y evaluación de cada *startup* (equipo tecnológico) de su progreso, manteniendo una comunicación constantes para finalizar en una exposición real de sus propuestas de valor ante un número de inversionistas reales.

ULA *Startups* está acorde con los nuevos tiempos académicos en que vivimos, donde los problemas de la sociedad en general, se pueden transformar en ideas de emprendimiento haciendo uso de tecnologías y conocimientos disponible. Todo esto implica un gran esfuerzo de organización y ejecución por lo que se requiere una plataforma tecnológica.

ULA *Startups* es la interfaz de entrada al ambiente de desarrollo que busca minimizar los

tiempos de desarrollo. Se requiere definir los ambientes de trabajo, desarrollo de todo el proceso de gerencia, organizar eventos y ponencias, construir los canales de comunicación entre las *Startups*, inversionistas y socios.

1.4 Objetivo General.

Diseñar, desarrollar e implementar un sistema de información web para la gestión de datos de ULA *Startups*.

1.4.1 Objetivos específicos.

www.bdigital.ula.ve

1. Estudiar y analizar en qué consiste ULA *Startups*, cual es el proceso de selección de una *startup* o emprendedor , y conocer las herramientas que se emplean en la aceleradora.
2. Estudiar y analizar de los requerimientos de la aceleradora para llevar a cabo la administración de la información y de los datos.
3. Definir los casos de usos.
4. Diseñar la estructura de datos a implementar.

C.C. Reconocimiento

5. Analizar y evaluar herramientas tecnológicas y módulos existentes que pueden ser útiles para el desarrollo de la plataforma para ULA *Startups* .
6. Desarrollar e implementar una interfaz de usuarios del sistema que permita una interacción amigable y dinámica.

www.bdigital.ula.ve

1.5 Metodología.

Se hace uso de la metodología de ciclos de desarrollo iterativo e incremental como *Scrum* [1] y *Lean Startup* [2] combinadas. Debido a las características del sistema de gestión para ULA *Startups* permite desarrollar una base funcional mínima y sobre ella ir incrementando o modificando las funcionalidades a desarrollar. Facilita la entrega continua de módulos para el tutor, lo que es una gran ventaja porque se pueden realizar pruebas sobre los módulos para comprobar su efectividad.

C.C. Reconocimiento

1.5.1 Recolección de historias de los usuarios.

Las historias de usuarios van a describir las funcionalidades y requisitos del software, son obtenidos directamente de los usuarios y se representan en frases pequeñas utilizando un lenguaje común [3].

Para este proyecto se cuenta con cuatro tipos de usuarios, los miembros de una *startups*, los cofundadores de ULA *Startups*, el administrador y los inversionistas.

www.bdigital.ula.ve

Luego de recolectar todas las historias de usuarios, se deben analizar las historias de valor para el desarrollo de software. Es de gran importancia considerar otras historias que no son expuestas, que pueden ser de gran importancia para el proyecto. Se deben asignar un valor desde el punto de vista del usuario, dar prioridad a algunas tareas con respecto a otras.

1.5.2 Pila del producto.

Es una lista ágil de los requisitos obtenidos en el paso anterior. Es importante priorizar las tareas de la lista, según lo obtenido en el paso anterior, y el estimado del nivel de esfuerzo por parte del desarrollador. El desarrollador debe leer la historia de usuario y subdividir en tareas de desarrollo de software que tendrán un valor, y que debe estimar según el esfuerzo o tiempo dedicado por el desarrollador. El nivel de esfuerzo se puede calcular de 1 a 13 según la serie de Fibonacci, es decir, de menor a mayor nivel de esfuerzo. Esta pila del proyecto nunca se da por completada está en continuo crecimiento y evolución. Al comenzar el proyecto incluye los requisitos iniciales y conforme avanza el desarrollo, y evoluciona el entorno en el que será usado, se van agregando nuevas tareas.

Es de gran utilidad como una herramienta para dirigir el desarrollo de cada módulo.

1.5.3 Iteraciones.

Es una técnica de desarrollar y entregar componentes incrementales de funcionalidades de un proyecto [6].

Luego de tener la pila del producto ordenada por la prioridad asignada por el tutor, se debe calcular el valor de esfuerzo estimado por el desarrollador, donde la suma de cada tarea no debe

superar los 25 puntos, para garantizar tiempos mínimos de finalización de la iteración.

De esta manera, cada iteración tendrá como resultado un producto mínimo viable para ser expuesto como entregas del proyecto. Con este método se busca tener una retroalimentación por parte del tutor y si hay aspectos o funcionalidades que modificar para mejorar el producto, se realizarán en el momento oportuno y no al final.

Es importante aclarar que cada Iteración cuenta con un producto que tendrá funcionalidades parciales del producto final.

www.bdigital.ula.ve

1.5.4 Análisis del producto mínimo viable.

Es la versión de un nuevo producto que permite a un equipo recolectar, con el menor esfuerzo posible, las posibles críticas de los clientes[10]. Este entregable con funcionalidades mínimas, producto de cada sprint será mostrado al tutor en cada entrega para ser evaluado y obtener sugerencias para nuevos cambios. Estos nuevos cambios a considerar serán agregados en la pila del producto con prioridades altas.

C.C. Reconocimiento

1.5.5 Retroalimentación.

Luego de obtener las sugerencias y cambios para el prototipo se inicia un proceso de reestructuración para cumplir con lo antes mencionado bajo los lineamientos de estabilidad y seguridad de manera que se cumpla con los nuevos requerimientos, evaluando si las primeras tareas se ajustan estos cambios sin perder su funcionalidad

1.6 Planificación.

www.bdigital.ula.ve

Tareas	Sem 1	Sem2	Sem 3	Sem 4	Sem 5	Sem 6	Sem 7	Sem 8
Recolección de historias de los usuarios	X							
Pila del producto	X							
Elaboración de las iteraciones.	X							
Iteración 1 (Diseño de estructura de datos)		X						

Análisis del producto mínimo viable			X					
Retroalimentación			X					
Iteración 2 (CRUD)				X				
Análisis del producto mínimo viable					X			
Retroalimentación					X			
Iteración 3 (vistas)						X		
Análisis del producto mínimo viable							X	
Retroalimentación								X

Tareas	Sem 1	Sem2	Sem 3	Sem 4	Sem 5	Sem 6	Sem 7	Sem 8
Recolección de historias de los usuarios	X							
Pila del producto	X							
Elaboración de las iteraciones.	X							

Iteración 1 (Diseño de estructura de datos)	X						
Análisis del producto mínimo viable		X					
Retroalimentación		X					
Iteración 2 (CRUD)			X				
Análisis del producto mínimo viable				X			
Retroalimentación				X			
Iteración 3 (vistas)					X		
Análisis del producto mínimo viable						X	
Retroalimentación							X

Capítulo 2

Especificaciones de ULA Startups.

2.1 Especificaciones de ULA Startups.

Se trata de reproducir la cultura de startups en el aula de clase, de incentivar a los alumnos a crear productos partiendo de una necesidad, con proyección a hacer un producto escalable. Esta nueva estrategia de desarrollo en el aula de clase tiene varias ventajas que vienen a solventar ciertas necesidades presentes en el aula de las universidades como:

- Existe una diferencia entre la paridad de la tecnología y la información presentada en las aulas. El alumno se distrae con gran facilidad por la falta de motivación en las materias. Muchos alumnos van en busca de nuevos retos aplicados a necesidades de la comunidad.
- La necesidad de sentirse en un ambiente de desarrollo a nivel laboral. Pero con la diferencia que van a emprender su propia idea lo cual le da un altísimo agregado. Pues mucho egresado de la ULA buscan un empleo y a cumplir horario para sentir luego una frustración de no sentirse realizados.

2.2 La estrategia de Plataforma web ULA *Startups*.

2.2.1 ULA startups desde el aula de clases.

En ULA STARTUPS, el alumno se enfocará en el desarrollo de un producto que viene a ser una solución a un problema. Esta solución se representará como un producto al finalizar el curso, donde además deberá aplicar los conocimientos adquiridos en el curso. Se busca promover una cultura de innovación y de crecimiento tanto académico como personal

2.2.2 ¿Cómo se implementa ULA startups?

Se simula el ambiente real de emprendimiento, los estudiantes deben organizarse en grupos que podrían simular startups. Estos equipos de trabajos por lo general deben estar conformados por un ceo cofundador y cofundadores, el ceo cofundador bien a hacer el responsable ante el profesor. Todos los integrantes deben aportar de igual forma y deben estar contagiados por el entusiasmo de dicho emprendimiento.

2.2 Especificaciones técnicas de la plataforma ULA startups.

Breve reseña de los fundamentos teóricos que hay detrás del diseño de la herramienta.

2.2.1 Plataforma web.

Una plataforma web es un sistema que permita administrar un sitio web. Haciendo que el sistema sea auto gestionable. [13]

2.2.1 Aceleradora de startups.

Una aceleradora de semillas o aceleradora de startups es una institución para impulsar startups mediante un programa basado en convocatorias con un plazo de tiempo estipulado. Estos programas incluyen mentorización, formación intensiva, educación digital y tutorización por parte de la empresa. [12]

www.bdigital.ula.ve

2.2.2 Arquitectura cliente servidor

La arquitectura cliente-servidor es un modelo distribuido en el que la tarea se reparten entre los proveedores de recursos o servicios, llamados servidores y los demandantes, llamados clientes . un cliente realiza peticiones a otro programa, el servidor quien le da respuesta. Esta idea también se puede aplicar a otro programa, el servidor, quien le da respuesta. Esta idea también se puede aplicar a programas que se ejecutan sobre una sola computadora, aunque es más ventajosa en un sistema operativo multiusuario distribuido a través de una red de computadoras.

2.2.4 Modelado de base de datos.

Arquitectura en capas El modelado de la base de datos es un tipo de modelo que determina la estructura lógica de una base de datos y de manera fundamental determina el modo de almacenar, organizar y manipular los datos.

Entre los modelos lógicos comunes para base de datos se encuentran:

Modelo jerárquico

Modelo en red

Modelo relacional

Modelo entidad-relación

Modelo entidad-relación extendido

Modelo de objetos

Modelo documental

Modelo entidad-atributo-valor

Modelo estrella

www.bdigital.ula.ve

Capítulo 3

Requerimientos y modelos

3.1 Requerimientos s del sistema.

Se hizo un estudio de todas las historias de los protagonistas y se recolecto la siguiente lista de requerimientos.

www.bdigital.ula.ve

3.1.1 Requerimientos funcionales.

- Gestión de Startups.
- Gestión de GeNetwork.
- Gestión de GeTeam.
- Registro de Ponencias .
- Registro de eventos.
- Gestión de categorías.
- Gestión de Inversionistas.
- Incorporar canales de comunicación.

- Registro de inversionistas.

3.1.2 Requerimientos no funcionales.

- El sistema de estar disponible para el usuario.
- El sistema debe poder dar acceso desde cualquier dispositivo del usuario.

3.1.3 Requerimientos de calidad.

- El sistema debe ser estable.
- El sistema debe ser intuitivo.
- El sistema de tener el mismo estilo en todas sus vistas o interfaces.

3.1.4 Requerimientos de seguridad.

- El sistema debe proporcionar autenticación a los usuarios de los distintos roles, administrador, ceo, cofundador de startup e inversionista.
- Asegurar la integridad de la información.

3.2 Modelado del sistema.

En base al Lenguaje Unificado de Modelado (UML), que prescribe un conjunto de notaciones y diagramas estándar para modelar sistemas orientados a objetos y describe la importancia de estos diagramas y símbolos.

ULA startups debe proveer una interfaz administrativa para el manejo de todo los integrantes y nuevas ponencias y eventos, tenemos nuestro primer actor tipo administrador. Para el manejo de las startup se contemplan dos actores el ceo como administrador absoluto de la startup, y el cofundador como observador y en algunas partes editor de la startup. Otro actor es el inversionista que debe ser capaz de ver las startup y de coordinar una reunión a través de la aceleradora para interactuar con dicha startup.

3.2.1 Actores.

Visitante: Es un actor que no ha identificado el sistema pero tiene acceso a la pagina principal del sistema. Para poder pertenecer al sistema deberá solicitar su registro.

Administrador: Representa la persona encargada de organizar eventos, ponencias, editar

startups, inversionistas, modificar categorías.

Cofundador CEO: Puede modificar todo lo que comprende su startup, desde su perfil, hasta sus integrantes.

Cofundador: Está al tanto de todos los aspectos de su startup y tiene ciertos privilegios para modificar algunas actividades.

Inversionista: Puede ver el listado de startup, estar al tanto de los próximos eventos, y solicitar reunión con alguna startup a través de la aceleradora.

www.bdigital.ula.ve

3.2.2 Entidades de negocios

Dentro del escenario se identifican las siguientes entidades de negocio, con las cuales pueden interactuar los actores.

Startup: La startup es el emprendimiento, la idea en plena ejecución. Representa las metas, objetivos y modelo de negocio de los emprendedores los cuales deben trabajar durante el curso para alcanzar un producto mínimo viable.

MVP: Es el producto mínimo viable que se obtendrá al final del proceso de desarrollo durante el curso. Que deberá cumplir las pautas del curso.

Inversión: Es la motivación a seguir con el producto para lograr obtener un producto completo y funcional, luego de haber pasado por las distintas etapas de retroalimentación de los usuarios de dicho producto.

Curso: Es el ambiente donde se va a desarrollar las startups y sus productos. Será el ambiente de aceleradora que va a proporcionar a los alumnos la motivación para desarrollar y complementar sus conocimientos con un emprendimiento.

www.bdigital.ula.ve

A continuación, se presentan los actores con las entidades de negocio y su interacción.

Evento: Es una reunión organizada por la aceleradora para exponer los MVP de sus startups a los potenciales inversionistas. Usualmente son los picos de evaluación durante el curso donde se desarrolla el MVP.

Ponencias: Forman parte del programa del desarrollo de MVP, se dan las herramientas, conocimientos y formación a los estudiantes, finalizando con invitados que desarrollaron su emprendimiento de manera exitosa. Se comparten experiencias y conocimientos. Y viene a ser

C.C. Reconocimiento

parte fundamental del curso.

Ahora la interacción de los actores con las entidades de negocios pueden verse en las siguientes figuras.

Para el administrador:

www.bdigital.ula.ve

C.C. Reconocimiento

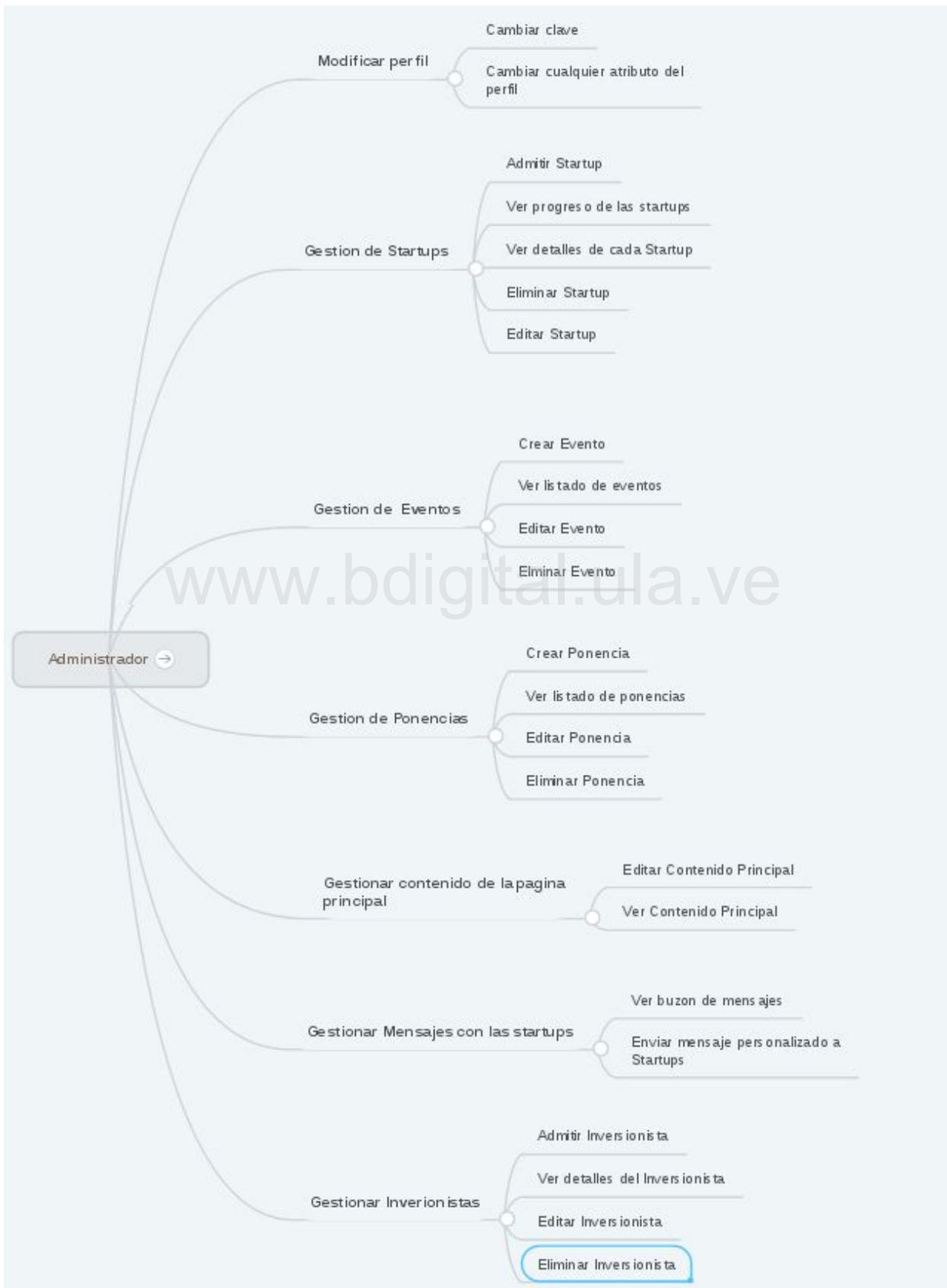


Figura 5. Posibilidades y restricciones para el administrador.

Para el cofundador:

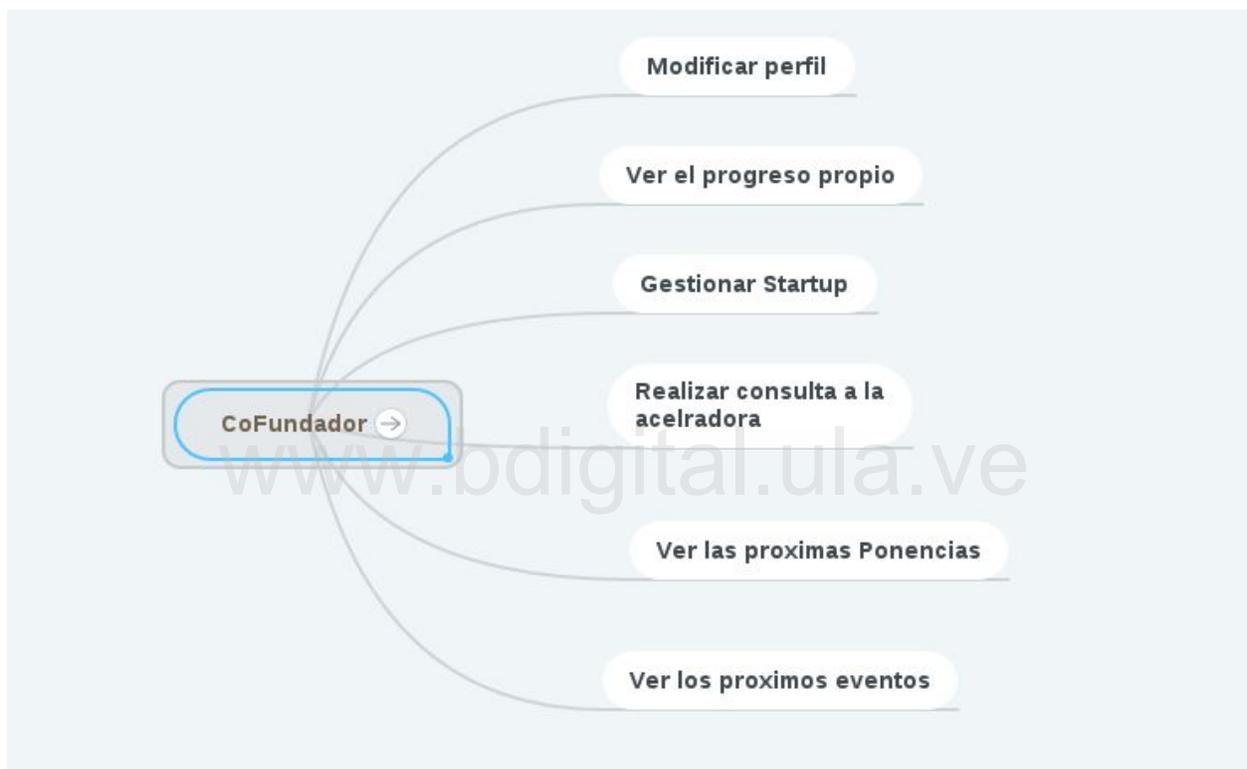


Figura 6. Posibilidades y restricciones para el cofundador de la Startup.

Para el inversionista:

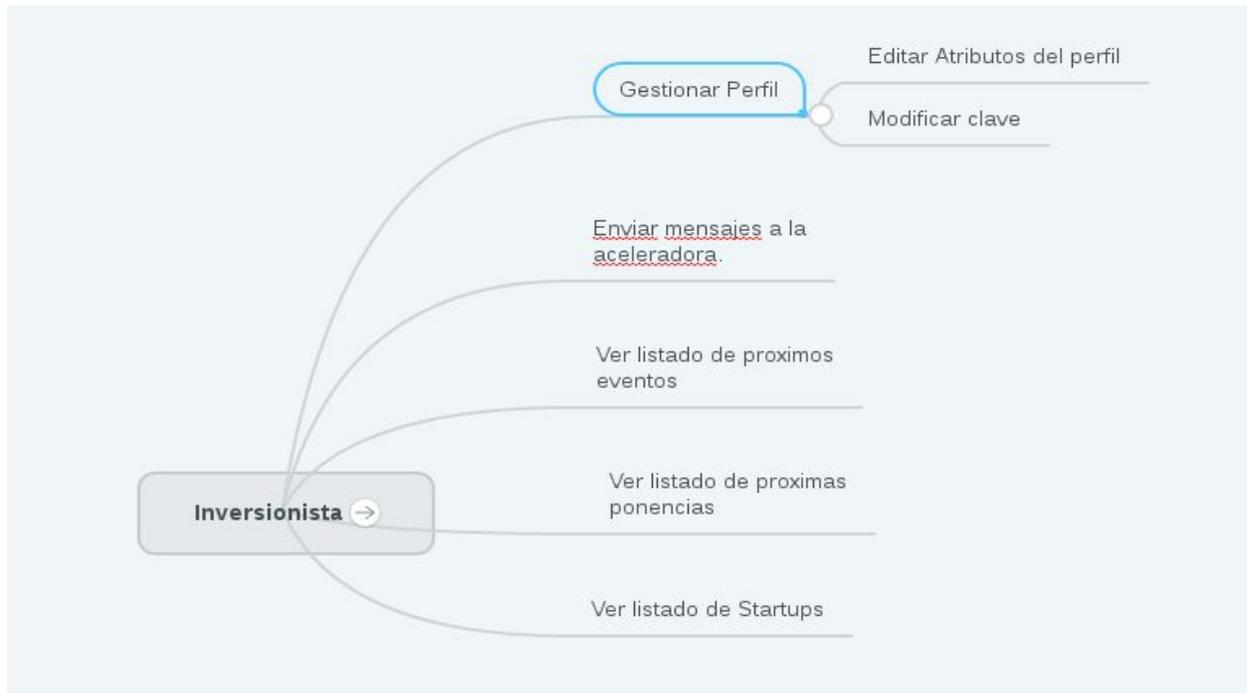


Figura 7. Posibilidades y requerimientos para el Inversionista

3.3 Casos de uso

A continuación se presentarán los casos de uso según los roles de los actores.

3.3.1 Actor Usuario

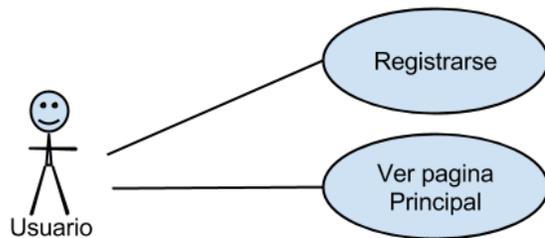


Figura 8. Casos de uso del usuario.

3.3.1.1 Registrar

Nombre:Registrar

Descripción:Permite que un usuario puede solicitar su admisión al sistema

Actores: Usuario

Precondiciones:

- El usuario no debe estar registrado en el sistema

Flujo Normal:

Ingresar datos:

- 1.- El actor presiona sobre el link para registrarse.
- 2.-El sistema presenta el formulario con los campos que el usuario debe llenar.
El usuario debe indicar si quieres ser cofundador de una startup o inversionista.
- 3.-El actor completa la ficha y presiona el botón Solicitar.
- 4.-El sistema valida el formulario.

Flujo Alternativo:

Crear:

- Los datos introducidos por el actor no son validados por el sistema, se le muestra el formulario mostrando en rojo los campos que el sistema no aprobó.

Precondiciones:

- La aceleradora es notificada de solicitud de admisión del nuevo usuario. La aceleradora decidirá si el usuario cumple con las políticas internas de admisión.

Tabla 1. Casos de uso: Registrar.

3.3.1.2 Ver pagina principal

Nombre: Ver página principal.

Descripción: Permite que un usuario no registrado pueda ver la página principal de la plataforma.

Actores: Usuario.

Precondiciones:

Ninguna

Flujo Normal:

Crear:

- 1.- El actor ingresa a la página principal del sistema a través del navegador.
- 2.- El sistema presenta la información general de la aceleradora.

Flujo Alternativo:

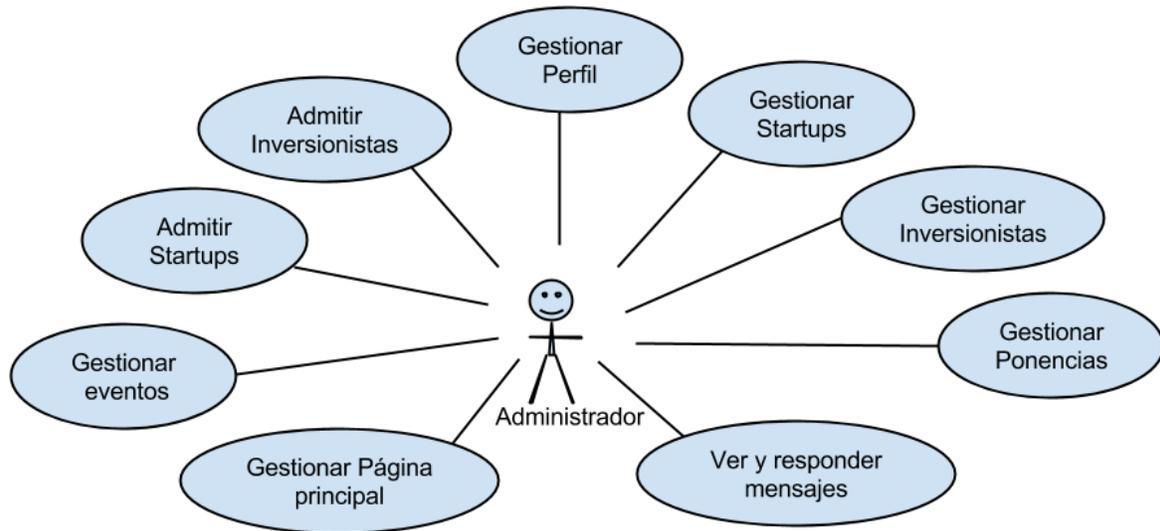
Ninguna

Precondiciones:

Ninguna

Tabla 2. Casos de uso: Ver página principal.

3.3.2 Actor Administrador.



www.bdigital.ula.ve

Figura 9. Casos de uso del administrador.

3.3.2.1 Cambiar perfil.

Nombre: Gestionar Perfil

Descripción: Permite que un usuario registrado como administrador pueda editar su clave, nombre y foto de perfil.

Actores: Administrador.

Precondiciones:

El usuario debe estar autenticado en el sistema.

Flujo Normal:

Crear:

- 1.- El actor presiona sobre el botón, del lado derecho superior, editar perfil.
- 2.-El sistema presenta la ficha con los campos que desea editar.
- 3.-El actor modifica los campos de interés.
- 4.-El sistema valida los campos ingresados por el actor.

Flujo Alternativo:

- Los datos introducidos por el actor no fueron validados por el sistema.El sistema mostrar los campos en rojo que no fueron validados.

Poscondiciones:

- Los datos fueron modificados.

Tabla 3. Casos de uso: Gestionar Perfil.

3.3.2.2 Gestionar startups.

Nombre:Gestionar startups.

Descripción:Permite que el administrador pueda ver un listado de startups, editar y eliminar.

Actores: Administrador

Precondiciones:

El usuario debe estar autenticado en el sistema

Flujo Normal:

Crear:

- 1.- El actor ingresa al listado de las startups a través del botón Gestionar Startups del panel.
- 2.-El sistema presenta la información general de cada startup en un listado incluyendo dos acciones sobre ellas editar o eliminar.
 - 2.1.-El actor selecciona la opción editar.
 - 2.1.1-El sistema mostrar los campos de la startup para editar.
 - 2.1.2-El actor selecciona el botón guardar.
 - 2.1.3-El sistema validará los campos.
 - 2.2- El actor selecciona el botón eliminar.

Flujo Alternativo:

- El sistema mostrará los campos no validados en rojo.

Poscondiciones:

Su Acción fue realizada con éxito.

Tabla 4. Casos de uso: Gestionar Startups.

3.3.2.3 Gestionar Inversionistas.

Nombre:Gestionar inversionistas.

Descripción:Permite que el administrador pueda ver un listado de inversionistas, editar y eliminar.

Actores: Administrador

Precondiciones:

El usuario debe estar autenticado en el sistema

Flujo Normal:

Crear:

- 1.- El actor ingresa al listado de las inversionistas a través del botón Gestionar Inversionistas del panel.
- 2.-El sistema presenta la información general de cada inversionista en un listado incluyendo dos acciones sobre ellas editar o eliminar.
 - 2.1.-El actor selecciona la opción editar.

- 2.1.1-El sistema mostrar los campos de la inversionista para editar.
- 2.1.2-El actor selecciona el botón guardar.
- 2.1.3-El sistema validará los campos.
- 2.2- El actor selecciona el botón eliminar.

Flujo Alternativo:

- El sistema mostrará los campos no validados en rojo.

Poscondiciones:

Su Acción fue realizada con éxito.

Tabla 5. Casos de uso: Gestionar Inversionistas.

3.3.2.4 Gestionar Ponencias.

Nombre:Gestionar Ponencias.

Descripción:Permite que el administrador pueda ver un listado de Ponencias,crear, editar y eliminar.

Actores: Administrador

Precondiciones:

El usuario debe estar autenticado en el sistema

Flujo Normal:

Crear:

- 1.- El actor ingresa al listado de las ponencias a través del botón Gestionar Ponencias del panel.
- 2.-El sistema presenta la información general de cada ponencia en un listado incluyendo dos acciones sobre ellas crear, editar o eliminar.
 - 2.1.-El actor selecciona la opción editar.
 - 2.1.1-El sistema mostrar los campos de la ponencia para editar.
 - 2.1.2-El actor selecciona el botón guardar.
 - 2.1.3-El sistema validará los campos.
 - 2.2- El actor selecciona el botón eliminar.
 - 2.3.-El actor selecciona la opción crear.
 - 2.3.1-El sistema los campos de la nueva ponencia que debe llenar.
 - 2.3.2-El actor selecciona el botón guardar.
 - 2.3.3-El sistema validará los campos.

Flujo Alternativo:

- El sistema mostrará los campos no validados en rojo.

Poscondiciones:

Su Acción fue realizada con éxito.

Tabla 6. Casos de uso: Gestionar Ponencias.**3.3.2.5 Gestionar Eventos.**

Nombre: Gestionar Eventos.

Descripción: Permite que el administrador pueda ver un listado de Eventos, crear, editar y eliminar.

Actores: Administrador

Precondiciones:

El usuario debe estar autenticado en el sistema

Flujo Normal:

Crear:

- 1.- El actor ingresa al listado de las ponencias a través del botón Gestionar Eventos del panel.

- 2.-El sistema presenta la información general de cada evento en un listado incluyendo dos acciones sobre ellas crear, editar o eliminar.
 - 2.1.-El actor selecciona la opción editar.
 - 2.1.1-El sistema mostrar los campos de la evento para editar.
 - 2.1.2-El actor selecciona el botón guardar.
 - 2.1.3-El sistema validará los campos.
 - 2.2- El actor selecciona el botón eliminar.
 - 2.3.-El actor selecciona la opción crear.
 - 2.3.1-El sistema los campos de la nueva evento que debe llenar.
 - 2.3.2-El actor selecciona el botón guardar.
 - 2.3.3-El sistema validará los campos.

Flujo Alternativo:

- El sistema mostrará los campos no validados en rojo.

Poscondiciones:

Su Acción fue realizada con éxito.

Tabla 7. Casos de uso: Gestionar Eventos.

3.3.2.6 Gestionar Pagina Principal.

Nombre: Gestionar Página principal.

Descripción: Permite que el administrador pueda ver el contenido mostrado en la página principal con la posibilidad de editar.

Actores: Administrador

Precondiciones:

El usuario debe estar autenticado en el sistema

Flujo Normal:

Crear:

- 1.- El actor ingresa al contenido de la página principal con opción a editar.
- 2.-El sistema presenta el contenido de la pagina principal
- 3.-El actor selecciona la opción editar.
- 4.-El sistema mostrar los campos de la sección para editar.
- 5.-El actor selecciona el botón guardar.
- 6.-El sistema validará los campos.

Flujo Alternativo:

- El sistema mostrará los campos no validados en rojo.

Poscondiciones:

Su Acción fue realizada con éxito.

Tabla 8. Casos de uso: Gestionar Página Principal.

3.3.2.7 Admitir Startup.

Nombre: Admitir Startup

Descripción: Permite que el administrador pueda admitir según su criterio a un nuevo usuario que desea pertenecer a la plataforma como startup

Actores: Administrador

Precondiciones:

El usuario debe estar autenticado en el sistema

Flujo Normal:

Crear:

- 1.- El actor ingresa al listado de notificaciones.
- 2.- El sistema presenta el listado de solicitudes.
- 3.- El actor selecciona la opción admitir o rechazar.

- 4.-El sistema Enviará un correo al usuario con el mensaje de admitido o no admitido.

Flujo Alternativo:

- Ninguna

Poscondiciones:

Su Acción fue realizada con éxito.

Tabla 9. Casos de uso: Admitir Cofundador.

3.3.2.8 Admitir Inversionista.

Nombre:Admitir Inversionista

Descripción:Permite que el administrador pueda admitir según su criterio a un nuevo usuario que desea pertenecer a la plataforma como startup

Actores: Administrador

Precondiciones:

El usuario debe estar autenticado en el sistema

Flujo Normal:

Crear:

- 1.- El actor ingresa al listado de notificaciones.
- 2.-El sistema presenta el listado de solicitudes.
- 3.-El actor selecciona la opción admitir o rechazar.
- 4.-El sistema Enviará un correo al usuario con el mensaje de admitido o no admitido.

Flujo Alternativo:

- Ninguna

Poscondiciones:

Su Acción fue realizada con éxito.

Tabla 10. Casos de uso: Admitir inversionista.**3.3.2.9 Ver y responder mensajes.**

Nombre: Ver y responder mensajes

Descripción: Permite que el administrador pueda ver un listado de mensajes y poder responderlos.

Actores: Administrador

Precondiciones:

El usuario debe estar autenticado en el sistema

Flujo Normal:

Crear:

- 1.- El actor ingresa al listado de mensajes.
- 2.-El sistema presenta el listado de mensajes.
- 3.-El actor selecciona la opción responder.
- 4.-El sistema muestra los campos para la respuesta
- 5.- El actor selecciona el botón enviar.
- 6.-El sistema valida el formulario y si esta mal marcar el campo en rojo

Flujo Alternativo:

- Ninguna

Poscondiciones:

Su Acción fue realizada con éxito.

Tabla 11. Casos de uso: Ver y responder mensajes.

Capítulo 4

Diseño de arquitectura.

4.1 Justificación y diseño.

La plataforma ULA startups necesita ser multiusuario, que sea de fácil acceso y proveer la información necesaria en el momento que el usuario así lo requiera. Entre las múltiples arquitecturas existentes a elegir la más apropiada para cumplir con estas funcionalidades es la arquitectura web. La arquitectura web ofrece todas las características necesarias para un funcionamiento adecuado a los requisitos antes mencionados.

Debido a la exigente necesidad de mostrar avances continuos y productos mínimos viables, las tecnologías de desarrollo de software deben proporcionar al programador mayores facilidades de desarrollo que permitan desarrollar las funcionalidades y de ser necesarias recurrir de nuevas tecnologías para completar los requisitos.

Para el desarrollo de la plataforma ULA startups se utilizaron las siguientes tecnologías.

- Para el manejador de base de datos se utilizó Mysql 5.1. Las ventajas con respecto a otros manejadores es que, soporta gran variedad de sistemas operativos, la velocidad al realizar las operaciones lo que lo hace uno de los gestores con mejor rendimiento, bajo costo en cuanto recursos que permita que sea ejecutado en cualquier máquina, facilidad de configuración e instalación, soporta gran variedad de sistemas operativos, baja probabilidad de corromper datos, entre otras características lo hacen uno de los gestores más recomendados en la actualidad [15].
- Para el desarrollo de la plataforma se utilizó Php 5.5, debido a la gran facilidad a la hora de desarrollar sistemas web, debido a que este lenguaje fue pensado entre otras cosas para el desarrollo de sistemas web. Otras de las razones por las que se seleccionó esta tecnología es porque cuenta con una extensa documentación y foros de desarrolladores que permiten resolver con gran facilidad cualquier contratiempo. Al momento de la instalación es sencillo y está disponible para diferentes sistemas operativos y servidores web por ser compatible con apache presente en la gran mayoría de los servidores.
- El Framework usado es Laravel 5.2, por contar con una documentación extensa y foros de desarrolladores. Laravel tiene soporte para múltiples drivers de autenticación,

generador de sistemas de autenticación, middleware para limitar el número de peticiones por dirección ip, posibilidad de agrupar varios middlewares, mejoras en la validación de arreglos de formularios. [14]

4.2 Diagrama de despliegue.

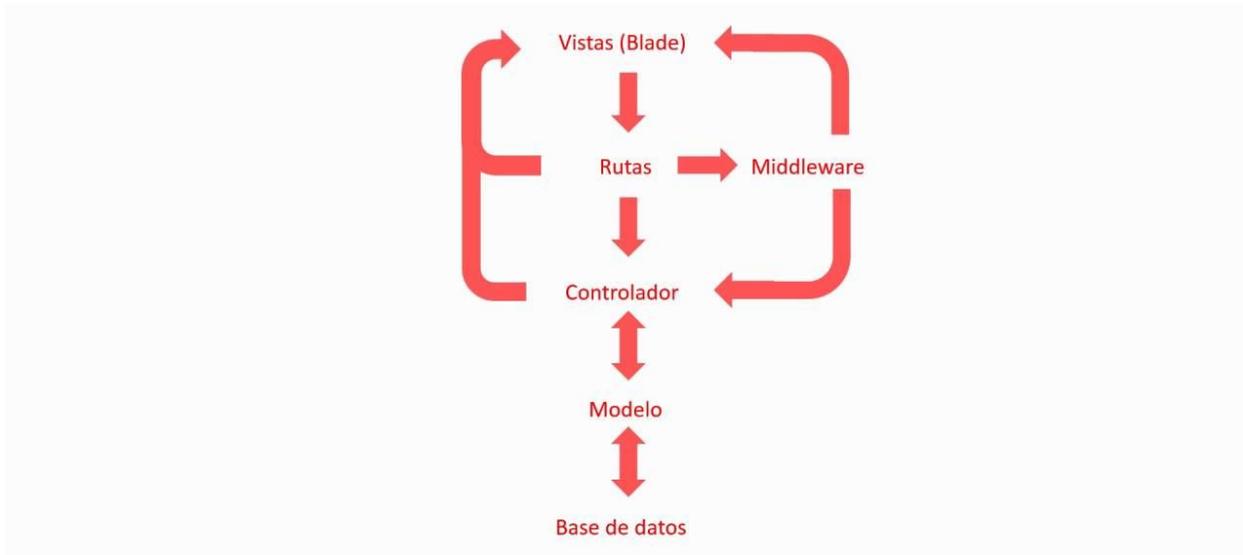
Para hacer uso de la plataforma ULA *Startups* es necesario tener la infraestructura expuesta en la figura 2. Diagrama de despliegue, de lado de cliente es necesario el uso de un navegador con soporte para javascript, del lado servidor es necesario un servidor http con capacidad para ejecutar PHP (Apache 2 y IIS son algunas alternativas) y un manejador de base de datos, Mysql.

4.3 Arquitectura de software.

El patrón de software que se usó para este proyecto fue el MVC o Modelo Vista Controlador, esta arquitectura es la usada por Laravel, para más información apéndice 1, Figura 20. Estas capas de laravel representan lo siguiente.

- El Modelo: Es la representación de la información con la cual el sistema opera, por lo tanto, gestiona todos los accesos a dicha información, desde consultas como actualizaciones, implementando también los privilegios de acceso que se hayan descrito en las especificaciones de la aplicación.
- Vista: Presenta el 'modelo' (información y *lógica de negocio*) en un formato adecuado para interactuar (usualmente la interfaz de usuario) por tanto requiere de dicho 'modelo' la información que debe representar como salida.
- Controlador: Responde a eventos (usualmente acciones del usuario) e invoca peticiones al 'modelo' cuando se hace alguna solicitud sobre la información (por ejemplo, editar un documento o un registro en una base de datos). [16]

La aplicación en laravel 5.2 poseen el siguiente flujo.



1. El index es la vista principal, inicializa la pagina principal a la espera de cualquier solicitud.
2. Routes.php, el enrutador, determina que controlador y método se está solicitando.
3. Existe una capa de seguridad para filtrar los datos enviados por el usuario con el fin de preservar la integridad de los datos.
4. El controlador ejecuta el método solicitado haciendo uso de los modelos, librerías, drivers, ayudantes, paquetes o scripts en caso de ser necesario.
5. La vista generada y enviada al cliente, si está activado el cache para el método.

4.4 Modelo relacional de la base de datos.

Para la plataforma ULA startups se diseñó un modelo de base de datos relacional que engloba a la plataforma y las herramientas. Debido a las consultas comunes entre varias tablas se modeló una base de datos para todo el sistema.

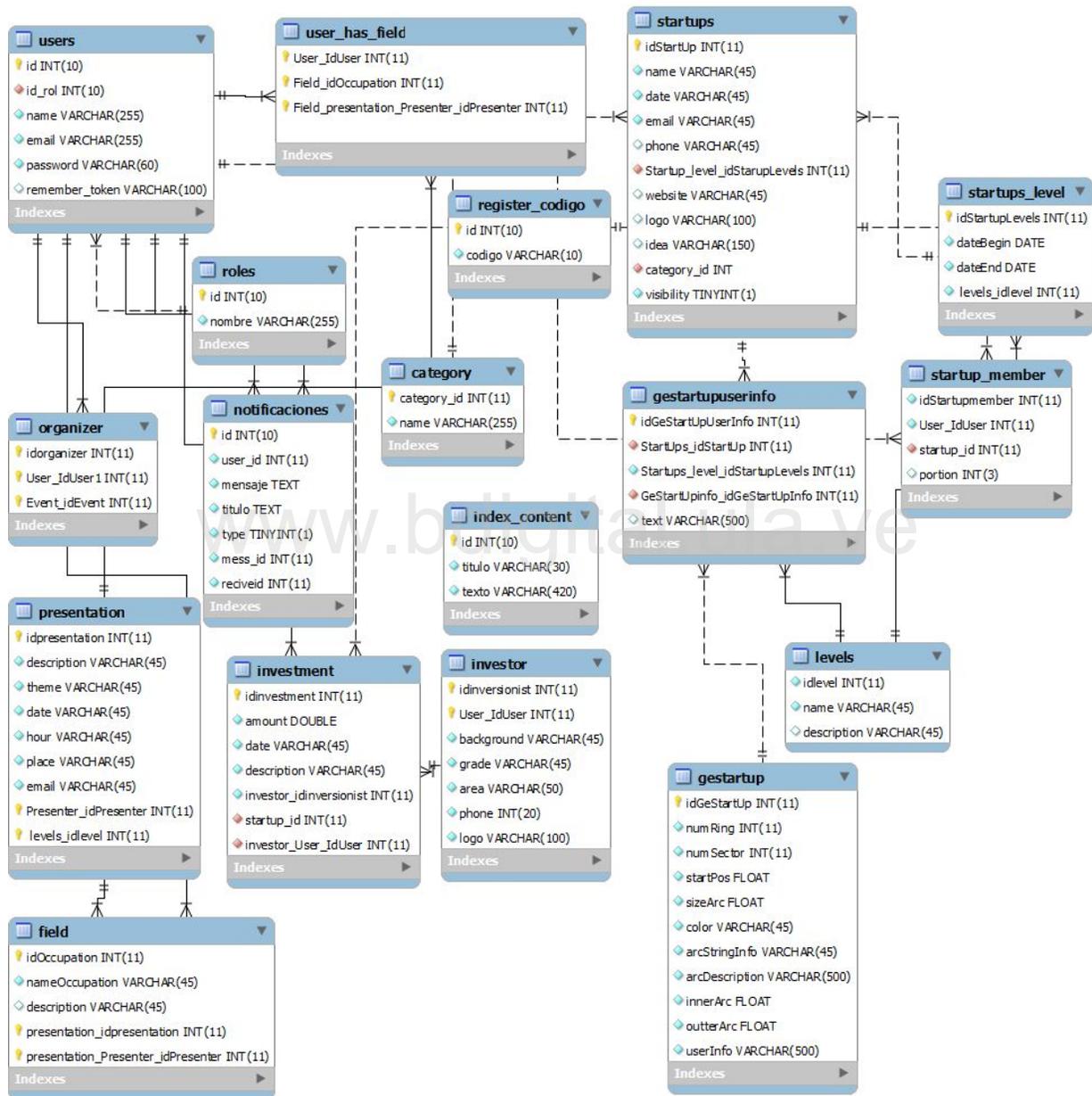
Para ver más detalles de las definiciones de la estructura de la base de datos en la sección de apéndice 1.

www.bdigital.ula.ve

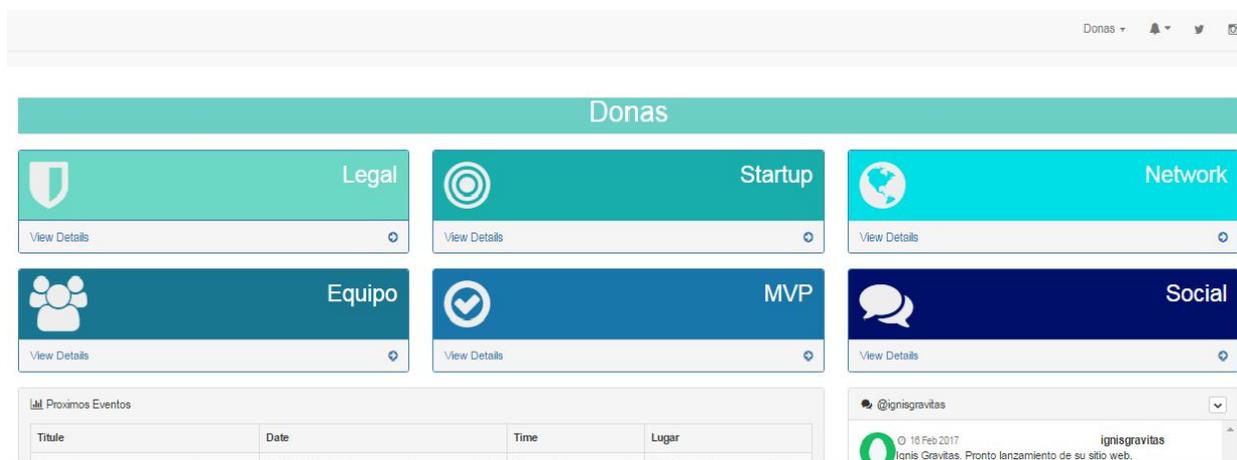
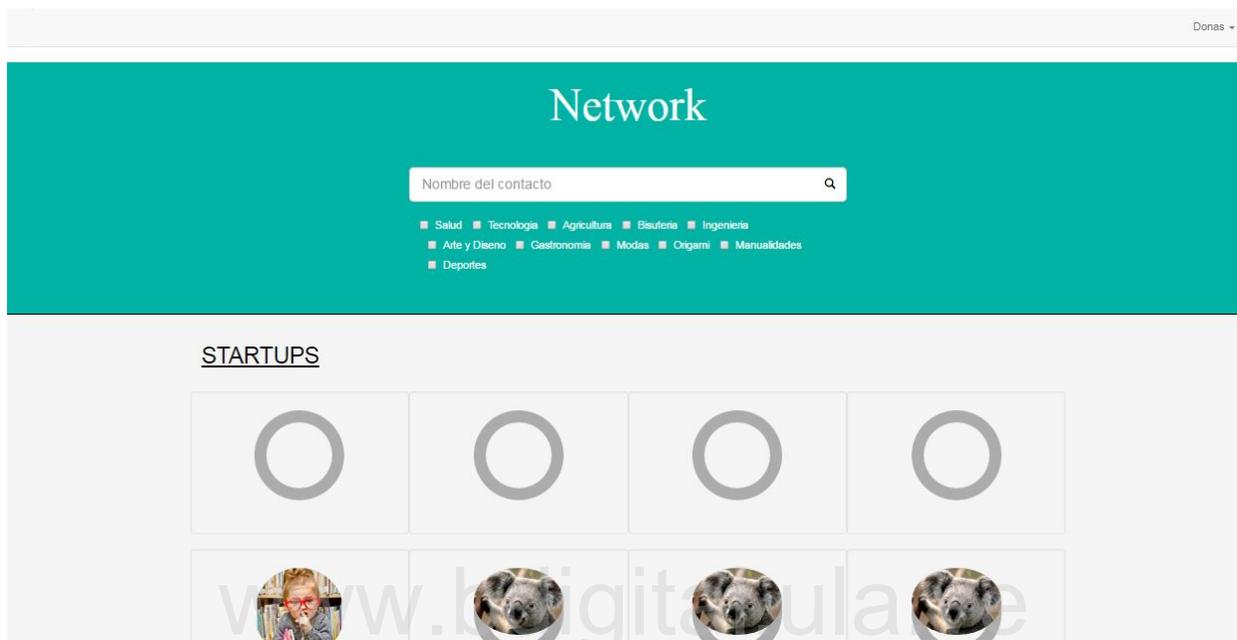
Se analizó las dependencias funcionales se puede determinar lo siguiente:

1. No existen en el modelo relacional atributos multi-evaluados y todas las tuplas tienen el mismo número de columnas, por consiguiente, está en primera forma normal.
2. Los atributos de todas las tablas dependen de la clave primaria y está en primera forma normal, por lo tanto, está en segunda forma normal.
3. No existen dependencias transitivas en el modelo y está en segunda forma normal, por lo tanto, está en tercera forma normal.

4.5 Diagrama de entidad relación extendido de la base de datos.



4.6 Diseño de la interfaz de usuarios



Capítulo 5

Especificaciones, desarrollo e implementación.

5.1 Desarrollo de ULA Startups.

Para el desarrollo de ULA Startups primero fue preparado el framework laravel instalando el composer con todas las librerías que se consideraron necesarias.

Para el control de versiones se uso git, este permite crar copias de la aplicación con fines de desarrollo, luego de cada modificaciones se hace commit para guardar esta versión con las nuevas modificaciones, teniendo un control del desarrollo en todas sus fases de desarrollo. Si durante las pruebas fallaba algo se podía volver al commit anterior y reanudar el desarrollo.

Para el desarrollo de ULA Startups fue preparado el framework, tomando como base Laravel,

que viene con una estructura de directorios bastante clara para el desarrollo, la arquitectura MVC. Se comienza haciendo todos los modelo de datos, de segundo la ruta, el controlador y por último la vista. Luego de tener claro cada una de las vistas se mecanizo el proceso, probando cada vista y que no diera error.

Los controladores contiene toda la lógica y funcionalidad de negocio, así como las validaciones de los datos provenientes de la capa vista. Estas se ubican en el directorio “raíz/app/Http/controllers” y son los siguientes:

- adminController.php
- InicioController.php
- CategoriasController.php
- InversionistaController.php
- CofundadorController.php
- JsonController.php
- CrearStartupController.php
- NetworkController.php
- EventoController.php
- NotificacionesController.php
- GestartupController.php
- PonenciaController.php
- HitosController.php
- AuthController.php
- HomeController.php
- RegisterController.php

De la lista anterior podemos destacar los siguiente controladores; inicioController.php que contiene la información mostrada en la página principal, AuthController.php que contiene el

mecanismo de autenticación del sistema y RegisterController.php maneja el registro por primera vez de un usuario.

El resto de los controladores son CRUD de tablas que corresponden a su mismo nombre, todas están compuestas por un método principal llamado index, que se trae toda la información mostrada en la vista, update para editar cada dato previamente seleccionado y modificado, delete para borrar el dato seleccionado, e insert para crear un nuevo dato.

Las vistas están en el directorio “raíz/resources/views” y son las siguientes:

En administrador:

www.bdigital.ula.ve

- Startup.blade.php
- Categoría.blade.php
- CrearStartup.blade.php
- Dashboard.blade.php
- Dashboarden.blade.php
- Evento.blade.php
- Index.blade.php
- Inversionista.blade.php
- Notificaciones.blade.php
- Ponencia.blade.php

5.2 Pruebas del sistema

Previo a la entrega de los prototipos, se realizaron pruebas unitarias durante el desarrollo. La aplicación fue resguardada a partir del Golden Model en un servidor de Google Cloud, de acceso privado para la realización de pruebas alfa por usuarios. Los usuarios que realizaron pruebas fue en el curso de sistemas computacionales. Cada uno representando un rol e indicando los errores o bugs que se presentaron.

Durante estas pruebas fueron modificados y corregidos la mayoría de los módulos ya que se mantenía la adición de requerimientos continuamente. Las pruebas realizadas se hicieron desde los navegadores Google Chrome, Opera y Firefox.

www.bdigital.ula.ve

C.C. Reconocimiento

www.bdigital.ula.ve

C.C. Reconocimiento

Capítulo 6.

Conclusiones y recomendaciones

6.1 Conclusiones

Luego de estudiar y analizar en qué consistía una aceleradora de Startups, cómo se implementaría en el salón de clases para llevar a cabo la dinámica de desarrollo bajo este ambiente de emprendimiento, y sabiendo que no una herramienta tecnológica que facilitará y llevará la gestión se pudo ver las deficiencias y restricciones que tenía esta nueva estrategia sin una herramienta que sirviera de apoyo. Esto permitió definir una serie de requerimientos tecnológicos para desarrollar un sistema de gestión y seguimiento que fuera óptimo.

De esta manera en este trabajo se diseñó, desarrolló, y validó una nueva herramienta tecnológica como plataforma de gestión ULA Startups, basada en una arquitectura web cliente servidor, que facilita las funciones de gestión, propias de un administrador, de un CEO cofundador, de un cofundador y de un inversionista, con canales de comunicación no presenciales (notificaciones) alcanzando el objetivo general de este trabajo.

A continuación, listamos los objetivos alcanzados.

Ambiente Administrador (profesor).

- ✓ Presentación del listado de startups, información básica como nombre, idea, estado global de progreso de cada uno, que producto tiene asignado e información de integrantes.
- ✓ Listado de inversionistas, con nombres, categoría, información de contacto.
- ✓ Listado de ponencias y eventos, con sus nombres, ponentes, fechas y lugar. Con la posibilidad de crear, editar y modificar.
- ✓ Listado de categorías presentadas para las startups e inversionistas.

Ambiente startup (Cofundador CEO)

- ✓ Se presenta un menú de acceso a todas las herramientas de la plataforma, genetwork, geteam, gemvp, gestartup,
- ✓ Listado de eventos y ponencias próximas.
- ✓ Últimos tweets de la aceleradora.
- ✓ Canal de comunicación con la aceleradora via notificación.
- ✓ La gestión de la información de la startup.

Ambiente inversionista

- ✓ Acceso a un listado de startups solo dando información puntual de la mismas.
- ✓ Listado de ponencias y eventos próximos.

- ✓ Canal de comunicación con la aceleradora vía notificaciones.

6.2 Recomendaciones.

La plataforma ULA Startups está orientada a un ambiente de emprendimiento que puede ser aplicado en la universidad. Se debe introducir al curso a un ambiente de desarrollo de emprendimiento. Es bueno definir el léxico utilizado en estos ambientes para facilitar un mejor entendimientos en los diferentes entornos.

La funcionalidad del léxico requiere de la traducción al inglés. La plataforma fue desarrollada en dos idiomas para lograr la internacionalización de la plataforma.

Capítulo 7.

Bibliografía.

[0] Fundadores de Telanto(2012). Telanto. Recuperado en el 10 de marzo del 2016 en <https://www.linkedin.com/company/telanto>.

[1] Colaboradores de Wikipedia (26 de febrero del 2016). Scrum. Recuperado el 1 de febrero del 2016 en <https://es.wikipedia.org/wiki/Scrum>.

[2] Colaboradores de Wikipedia (27 de noviembre del 2015). Lean startup. Recuperado el 17 de febrero del 2016 en https://es.wikipedia.org/wiki/Lean_startup.

[3] Colaboradores de Wikipedia (27 de octubre del 2015). Historia de usuarios. Recuperador el 15 de octubre del 2015 en https://es.wikipedia.org/wiki/Historias_de_usuario.

[5] Colaboradores de Scrummanager(16 de junio del 2014). Pila del producto. Recuperado el 18 de marzo del 2014 http://www.scrummanager.net/bok/index.php?title=Pila_del_producto.

[6] Colaboradores de Wikipedia (15 de marzo del 2016). Iteración. Recuperado el 10 de marzo del 2016 <https://es.wikipedia.org/wiki/Iteraci%C3%B3n>.

[10] Colaboradores de Wikipedia(7 de enero de 2013). Producto viable mínimo. Recuperado el

[11] 17 de febrero de 2016, de https://es.wikipedia.org/wiki/Producto_viable_m%C3%ADnimo.

[12] Colaboradores de Wikipedia(12 de octubre de 2016). Aceleradora de Startups. Recuperado el 22 de enero de 2017,https://es.wikipedia.org/wiki/Aceleradora_de_startups.

[13] Colaboradores de ALEGSA (16 de junio del 2009). Plataforma Web. Recuperado el 22 de enero de 2017 <http://www.alegsa.com.ar/Diccionario/C/3641.php>.

[14] Colaboradores de <https://styde.net/blog/> (22 de diciembre del 2015). Laravel 5.2. Recuperado el 24 de enero del 2017 <https://styde.net/laravel-5-2-ya-esta-disponible/>.

[15] Colaboradores de <https://mysqldaniel.wordpress.com/> (21 de mayo del 2015). Ventajas de

Mysql. Recuperado el 25 de enero de 2017

<https://mysqldaniel.wordpress.com/ventajas-y-desventajas/>

[16] Colaboradores de <https://es.wikipedia.org> (20 de marzo del 2015). Modelo Vista

Controlador. Recuperado el 24 de enero del 2017

<https://es.wikipedia.org/wiki/Modelo%E2%80%93vista%E2%80%93controlador>

www.bdigital.ula.ve

C.C. Reconocimiento

Capítulo 8.

Apéndice

Glosarios y definiciones.

8.1 Definiciones técnicas.

www.bdigital.ula.ve

El Sistema a desarrollar va a estar funcionando bajo la arquitectura cliente-servidor, la cual consiste en una aplicación cliente que hace peticiones a una segunda aplicación servidor.

A continuación se definirán los conceptos necesarios para entender los detalles del desarrollo y funcionamiento del sistema a elaborar.

8.1.1 HTML

La siglas en inglés de *HyperText Markup Language* (lenguaje de marcas de hipertexto), hace

referencia al lenguaje de marcado para la elaboración de páginas web. Es un estándar que sirve de referencia del software que conecta con la elaboración de páginas web en sus diferentes versiones, define una estructura básica y un código (denominado código HTML) para la definición de contenido de una página web, como texto, imágenes, videos, juegos, entre otros.

8.1.2 Navegador Web.

Un navegador web es un software, aplicación o programa que permite el acceso a la Web, interpretando la información de distintos tipos de archivos y sitios web para que estos puedan ser visualizados.

www.bdigital.ula.ve

8.1.3 Javascript.

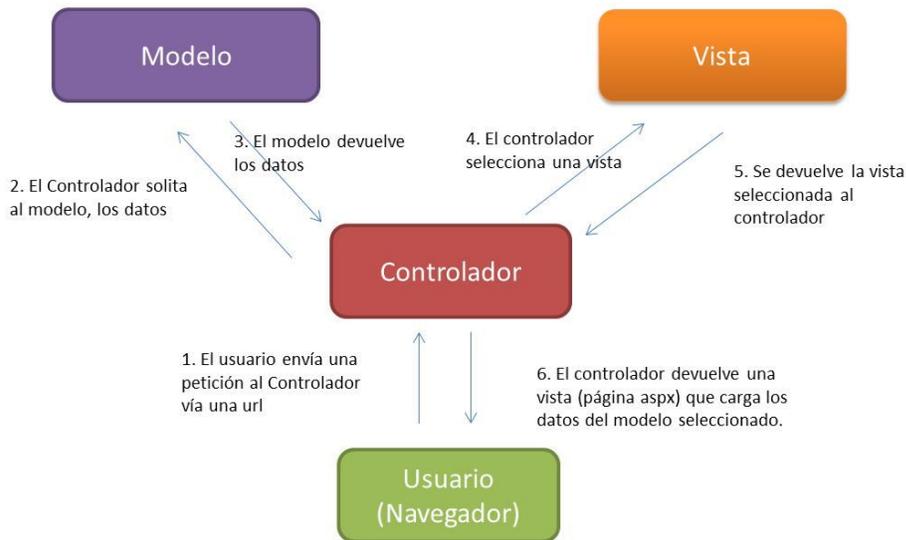
JavaScript (abreviado comúnmente JS) es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico.

8.1.4 PHP.

PHP (acrónimo recursivo de *PHP: Hypertext Preprocessor*) es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML.

8.1.5 MVC.

Modelo–vista–controlador (MVC) es un patrón de arquitectura de software, que separa los datos y la lógica de negocio de una aplicación de la interfaz de usuario y el módulo encargado de gestionar los eventos y las comunicaciones.



8.1.6 Framework.

www.bdigital.ula.ve

Un *framework*, entorno de trabajo o marco de trabajo es un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular que sirve como referencia, para enfrentar y resolver nuevos problemas de índole similar.

8.1.7 UML.

El lenguaje unificado de modelado (**UML**, por sus siglas en inglés, Unified Modeling Language) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad;

está respaldado por el Object Management Group (OMG).

8.1.8 SQL.

SQL (por sus siglas en inglés Structured Query Language; en español lenguaje de consulta estructurada) es un lenguaje específico del dominio que da acceso a un sistema de gestión de bases de datos relacionales que permite especificar diversos tipos de operaciones en ellos.

8.1.9 MySQL.

MySQL es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual GPL/Licencia comercial por Oracle Corporation y está considerada como la base datos open source más popular del mundo, y una de las más populares en general junto a Oracle y Microsoft SQL Server, sobre todo para entornos de desarrollo web.